



(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
06.05.2015 Bulletin 2015/19

(51) Int Cl.:
H04L 12/743^(2013.01) H04L 12/745^(2013.01)
H04L 12/701^(2013.01)

(21) Application number: **14189968.2**

(22) Date of filing: **22.10.2014**

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR
Designated Extension States:
BA ME

(72) Inventors:
• **Solis, Ignacio**
South San Francisco, CA California 94080 (US)
• **Mosko, Marc E.**
Santa Cruz, CA California 95060 (US)

(30) Priority: **31.10.2013 US 201314069251**

(74) Representative: **Lord, Michael**
Gill Jennings & Every LLP
The Broadgate Tower
20 Primrose Street
London EC2A 2ES (GB)

(71) Applicant: **Palo Alto Research Center Incorporated**
Palo Alto, California 94304 (US)

(54) **EXPRESS HEADER FOR PACKETS WITH HIERARCHICALLY STRUCTURED VARIABLE-LENGTH IDENTIFIERS**

(57) One embodiment of the present invention provides a system for forwarding named packets. During operation, the system receives a named packet with an express header. The named packet includes a payload and a name which identifies the payload. The express header includes a content identifier, a global forwarding identifier, and a local forwarding identifier. The content identifier uniquely represents an interest corresponding to the name. The global forwarding identifier includes forwarding information to facilitate global forwarding. The local forwarding identifier includes forwarding information to facilitate local forwarding. The system then makes a forwarding decision for the packet based on the content identifier, the global forwarding identifier, and optionally the local forwarding identifier.

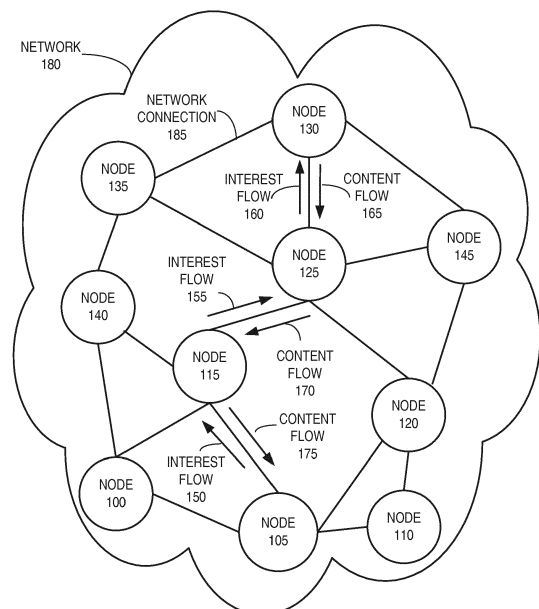


FIG. 1

Description

BACKGROUND

Field

[0001] The present disclosure relates generally to facilitating communication over a data network. More specifically, the present disclosure relates to a system and method for facilitating express header for packets with hierarchically structured variable-length identifiers.

Related Art

[0002] The proliferation of the Internet and e-commerce continues to fuel revolutionary changes in the network industry. Today, a significant number of information exchanges, from online movie viewing to daily news delivery, retail sales, and instant messaging, are conducted online. An increasing number of Internet applications are also becoming mobile. However, the current Internet operates on a largely location-based addressing scheme. The two most ubiquitous protocols, the Internet Protocol (IP) and Ethernet protocol, are both based on location-based addresses. That is, a consumer of content can only receive the content by explicitly requesting the content from an address (e.g., IP address or Ethernet media access control (MAC) address) closely associated with a physical object or location. This restrictive addressing scheme is becoming progressively inadequate for meeting the ever-changing network demands.

[0003] Recently, content centric network (CCN) architectures have been proposed in the industry. CCN brings a new approach to content transport. Instead of having network traffic viewed at the application level as end-to-end conversations over which content travels, content is requested or returned based on its unique name, and the network is responsible for routing content from the provider to the consumer. Note that content includes data that can be transported in the communication system, including any form of data such as text, images, video, and/or audio. A consumer and a provider can be a person at a computer or an automated process inside or outside the CCN. A piece of content can refer to the entire content or a respective portion of the content. For example, a newspaper article might be represented by multiple pieces of content embodied as data packets. A piece of content can also be associated with meta-data describing or augmenting the piece of content with information such as authentication data, creation date, content owner, etc.

[0004] In CCN, content objects and interests are identified by their names, which is typically a hierarchically structured variable-length identifier (HSVLI). Because these names have variable lengths, it is difficult to forward packets with HSVLIs at line speed with high throughput.

SUMMARY

[0005] One embodiment of the present invention provides a system for forwarding named packets. During operation, the system receives a named packet with an express header. The named packet includes a payload and a name which identifies the payload. The express header includes a content identifier, a global forwarding identifier, and a local forwarding identifier. The content identifier uniquely represents an interest corresponding to the name. The global forwarding identifier includes forwarding information to facilitate global forwarding. The local forwarding identifier includes forwarding information to facilitate local forwarding. The system then makes a forwarding decision for the packet based on the content identifier, the global forwarding identifier, and optionally the local forwarding identifier.

[0006] In a variation on this embodiment, the name is one of: a hierarchically structured variable-length identifier (HSVLI), a set of attributes, a flat identifier, an array of bytes, an entire set of data, or a portion of the data.

[0007] In a variation on this embodiment, the packet is an interest in a piece of content corresponding to the name. The content identifier is a hash derived based on the entire name and optionally additional information in the packet. The global forwarding identifier is a hash derived on the subset of the name components.

[0008] In a variation on this embodiment, the name used to calculate the global forwarding identifier or the local forwarding identifier is not structured.

[0009] In a variation on this embodiment, the system obtains the global forwarding identifier from a node providing a directory service.

[0010] In a variation on this embodiment, the system computes the global forwarding identifier based on the name or content of the packet.

[0011] In a variation on this embodiment, the system computes the local forwarding identifier based on the name or content of the packet.

[0012] In a variation on this embodiment, the name is an HSVLI. The system computes the global forwarding identifier by hashing a predetermined number of components of the HSVLI or a number of components of the HSVLI based on each component's type.

[0013] In a variation on this embodiment, the name is an HSVLI. The system computes the local forwarding identifier by hashing a predetermined number of components of the HSVLI or a number of components of the HSVLI based on each component's type.

[0014] In a variation on this embodiment, the system maintains a pending interest table, wherein a respective entry in the pending interest table indicates a pending interest associated with the content identifier, global forwarding identifier, and local forwarding identifier.

[0015] In a variation on this embodiment, the system maintains a forwarding information base, wherein a respective entry in the forwarding information base indicates forwarding information for a packet that contains a

global forwarding identifier.

[0016] In a further variation, the entry in the forwarding information base further indicates that local forwarding information exists corresponding to the packet's local forwarding identifier.

[0017] In a variation on this embodiment, the packet is an interest in the piece of content. In addition, the system searches a local content store for content corresponding to the interest based on the content identifier and optionally the global forwarding identifier and local forwarding identifier.

BRIEF DESCRIPTION OF THE FIGURES

[0018]

FIG. 1 illustrates an exemplary network where packets have hierarchically structured variable-length identifiers (HSVLIs) in accordance with an embodiment.

FIG. 2 illustrates an exemplary express header for a CCN packet, in accordance with an embodiment of the present invention.

FIG. 3 illustrates the process of forwarding an Interest and a corresponding Content Object with express headers, in accordance with an embodiment of the present invention.

FIG. 4 presents a flow chart illustrating the process of receiving and forwarding an Interest with an express header, in accordance with an embodiment of the present invention.

FIG. 5 illustrates an exemplary forwarding information base (FIB), in accordance with an embodiment of the present invention.

FIG. 6 illustrates an exemplary pending Interest table (PIT), in accordance with an embodiment of the present invention.

FIG. 7 presents a flow chart illustrating an exemplary process of receiving and forwarding a Content Object with an express header, in accordance with an embodiment of the present invention.

FIG. 8 illustrates an exemplary system for forwarding packets with HSVLIs and express headers, in accordance with an embodiment.

In the figures, like reference numerals refer to the same figure elements.

DETAILED DESCRIPTION

Overview

[0019] Embodiments of the present invention provide a system and method for using an express header, which includes fixed size, flat byte strings, to forward CCN packets with Hierarchically Structured Variable Length Identifiers (HSVLIs), thus simplifying the work done at a packet forwarder. A CCN express header includes a first byte

string, referred to as the content identifier (CI), a second byte string, referred to as the global forwarding identifier (GFI), and a third byte string, referred to as the local forwarding identifier (LFI).

[0020] The CI uniquely identifies a piece of content, which can correspond to an Interest or a Content Object. The CI remains invariant as a packet moves through the network.

[0021] The GFI represents forwarding information that can be used by global forwarders (such as core routers) to forward a CCN interest packet. Typically, the GFI can be associated with the longest matching prefix in the routing tables in various global forwarding devices (e.g., routers, switches, etc.) along a data path. It is assumed that GFI contains sufficient information for a core forwarder to forward a CCN Interest packet to a local network domain which hosts the requested content. The GFI might not contain all the information necessary for a forwarder to forward a CCN Interest to the final destination end system that hosts the content, but is sufficient for the core forwarders to forward the Interest packet to a network domain where the content resides.

[0022] The LFI represents more detailed forwarding information that can be used by a local forwarder (such as an enterprise router) to forward a CCN interest packet to the end system that hosts the requested content, once the packet reaches the destination network domain (as a result of being forwarded by core forwarders based on the GFI).

[0023] In essence, the CCN express header provides (1) a fixed-size byte string (CI) that can uniquely identify a piece of content, which can be used for a forwarder to locate the correct, cached Content Object to satisfy the interest; and (2) a fixed-size, two-tier data structure to carry forwarding information, which can be used by the core forwarders as well as local forwarders to efficiently forward the CCN.

[0024] A Content Object, sent in response to an Interest, carries the same express header as the corresponding Interest (i.e., the same CI, GFI, and LFI) along the return path so the Content Object may be forwarded along the proper path.

[0025] In general, CCN uses two types of messages: Interests and Content Objects. An Interest carries the hierarchically structured variable-length identifier (HSVLI), also called the "name," of a Content Object and serves as a request for that object. If a network element (e.g., router) receives multiple interests for the same name, it may aggregate those interests. A network element along the path of the Interest with a matching Content Object may cache and return that object, satisfying the Interest. The Content Object follows the reverse path of the Interest to the origin(s) of the Interest. A Content Object contains, among other information, the same HSVLI, the object's payload, and cryptographic information used to bind the HSVLI to the payload.

[0026] The terms used in the present disclosure are generally defined as follows (but their interpretation are

not limited to such):

"HSVLI": Hierarchically structured variable length identifier, also called a Name. It is an ordered list of Name Components, which may be variable length octet strings. In human-readable form, it can be represented in a format such as ccnx:/path/part. There is not a host or query string. As mentioned above, HSVLIs refer to content, and it is desirable that they be able to represent organizational structures for content and at least partially meaningful to humans. An individual component of a HSVLI may have an arbitrary length. Furthermore, HSVLIs can have explicitly delimited components, can include any sequence of bytes, and are not limited to human-readable characters. A longest-prefix-match lookup is important in forwarding packets with HSVLIs. For example, an HSVLI indicating an interest in "/parc/home/bob" will match both "/parc/home/bob/test.txt" and "/parc/home/bob/bar.txt." The longest match, in terms of the number of name components, is considered the best because it is the most specific.

"Interest": A request for a Content Object that specifies a HSVLI name prefix and other optional selectors to choose among multiple objects with the same name prefix. Any Content Object whose name matches the Interest name prefix and selectors satisfies the Interest. An interest packet is a packet that contains a CCN Interest.

"Content Object": A data object sent in response to an Interest. It has a HSVLI name and a Contents payload that is bound together via a cryptographic signature. Optionally, all Content Objects have an implicit terminal name component made up of the SHA-256 digest of the Content Object. In one embodiment, the implicit digest is not transferred on the wire, but is computed at each hop, if needed.

"Content Identifier": In an Interest, the Name and several fields called Selectors limit the possible content objects that match the interest. Taken together, they uniquely identify the query in the Interest. The Content Identifier is a fixed-size byte string that represents these fields. In one embodiment, the CI can be a hash of these fields. Two interests with the same CI are considered identical queries.

[0027] As mentioned before, an HSVLI indicates a piece of content, is hierarchically structured, and includes contiguous components ordered from a most general level to a most specific level. The length of a respective HSVLI is not fixed. In content-centric networks, unlike a conventional IP network, a packet may be identified by an HSVLI. For example, "abcd/bob/papers/ccn/news" could be the name of the content and identifies the corresponding packet(s); i.e., the "news" article from the "ccn" collection of papers for a user named "Bob" at the organization named "ABCD." To request a piece of content, a node expresses (e.g., broadcasts) an Interest in

that content by the content's name. An Interest in a piece of content can be a query for the content according to the content's name or identifier. The content, if available in the network, is routed back to it from any node that stores the content. The routing infrastructure intelligently propagates the Interest to the prospective nodes that are likely to have the information and then carries available content back along the path which the interest traversed.

[0028] FIG. 1 illustrates an exemplary architecture of a network, in accordance with an embodiment of the present invention. In this example, a network 180 comprises nodes 100-145. Each node in the network is coupled to one or more other nodes. Network connection 185 is an example of such a connection. The network connection is shown as a solid line, but each line could also represent sub-networks or super-networks, which can couple one node to another node. Network 180 can be content-centric, a local network, a super-network, or a sub-network. Each of these networks can be interconnected so that a node in one network can reach a node in other networks. The network connection can be broadband, wireless, telephonic, satellite, or any type of network connection. A node can be a computer system, an end-point representing users, and/or a device that can generate interests or originate content.

[0029] In accordance with an embodiment of the present invention, a consumer can generate an Interest in a piece of content and then send that Interest to a node in network 180. The piece of content can be stored at a node in network 180 by a publisher or content provider, who can be located inside or outside the network. For example, in FIG. 1, the Interest in a piece of content originates at node 105. If the content is not available at the node, the Interest flows to one or more nodes coupled to the first node. For example, in FIG. 1, the Interest flows (interest flow 150) to node 115, which does not have the content available. Next, the Interest flows (interest flow 155) from node 105 to node 125, which again does not have the content. The Interest then flows (interest flow 160) to node 130, which does have the content available. The flow of the content then retraces its path in reverse (content flows 165, 170, and 175) until it reaches node 105, where the content is delivered. Other processes such as authentication can be involved in the flow of content.

[0030] In network 180, any number of intermediate nodes (nodes 100-145) in the path between a content holder (node 130) and the Interest generation node (node 105) can participate in caching local copies of the content as it travels across the network. Caching reduces the network load for a second subscriber located in proximity to other subscribers by implicitly sharing access to the locally cached content

Forwarding CCN Packets based on Express Header

[0031] In general, a CCN packet, either for an Interest or Content Object, has a express header that includes a

content identifier (CI), a global forwarding identifier (GFI), and a local forwarding identifier (LFI). CI is used to uniquely identify a piece of content, and can be a hash of the name and one or more fields in the Interest packet. In one embodiment, CI is only computed by the source node that initiates an Interest, and optionally verified by an authoritative source node generating content or responding from a long-term repository. Any two Interests containing the same CI are considered to contain a request for the same piece of content. Any Content Object packet that contains the same CI is considered to be a correct response to the corresponding Interest. In essence, CI can be used in place of the name for purposes of identifying a piece of content.

[0032] The GFI can be computed based on or derived from one or more components of an Interest packet's name. In general, it is assumed that GFI contains sufficient forwarding information for the network to deliver an Interest packet to the network domain (such as an autonomous system or routing domain), in which at least one forwarding device has knowledge of how to deliver the Interest packet to its destination based on its LFI (described in more detail below).

[0033] In one embodiment, the GFI can be a hash of a certain number of top-level components of a CCN name, such as "/parc" in the name "/parc/lab/ccn/document." In other embodiments, the GFI can be an address (such as an IPV6 address) pointing to a destination network domain.

[0034] There are a number of ways for an end system to obtain the GFI for an Interest. In one embodiment, an end system can query another entity with the HSVLI to obtain the correct GFI. This other entity can be a device providing a DNS-like service for resolving HSVLIs to their respective GFIs. In another embodiment, an end system can use a predetermined convention to derive the GFI from an HSVLI. For example, it can be agreed among all nodes that all the top two components of a CCN name would be used for GFI computation. The end system can then use a hash of the top two components (such as "/parc/lab" in "/parc/lab/ccn/document") as its GFI. In a further embodiment, each component in a CCN name maybe designated with a type, and certain components in a CCN name can be of the type associated with global forwarding. For example, "/parc" and "/lab" maybe of the type associated with global forwarding in the name "/parc/lab/ccn/document." As a result, an end system initiating an Interest in "/parc/lab/ccn/document" can use the hash of "/parc/lab" as its GFI.

[0035] The LFI is similar to GFI, but with more specific forwarding information that can be used by a local forwarder (such as an enterprise router) that can forward an Interest packet to the correct destination node based on the LFI. For example, the LFI for an Interest with the HSVLI "/parc/lab/ccn/document" can be a hash of "/parc/lab/ccn" or "/parc/lab/ccn/document." In a further embodiment, each component in a CCN name may be designated with a type, and certain components in a CCN

name can be of the type associated with local forwarding. For example, "/parc" and "/lab" may be of the type associated with global forwarding in the name "/parc/lab/ccn/document" and "/ccn" can be associated with local forwarding. As a result, an end system initiating an Interest in "/parc/lab/ccn/document" can use the hash of "/parc/lab/ccn" or just the hash of "/ccn" as its LFI.

[0036] Note that there can be more than one LFI. In addition, the multiple LFIs appear in an ordered list and are evaluated in order as the respective network domains are reached.

[0037] On a high level, the CCN forwarding based on express headers operates as follows. A node issues an Interest for a Content Object and receives back at most one Content Object per Interest it sends. The Content Object's name is expected to be equal to or at least match a suffix of the Interest name, and to satisfy the various selectors in the Interest. In embodiments of the present invention, the system speeds up this processing by pre-computing the CI, GFI, and LFI. These three identifiers do not change in-route, and intermediate nodes do not need to do much expensive longest match for CCN flat-names. In particular, a forwarder does not necessarily evaluate the name or selectors when matching content in its Content Store (which serves as a cache for previously seen Content Objects). It may use exact match on the CI or exact match on the CI, GFI and LFI..

[0038] A forwarder typically maintains several data structures: The Pending Interest Table (PIT) tracks outstanding Interests the forwarder has seen, for which the forwarder is awaiting a response. It also aggregates similar Interests (Interest with the same CI and optionally the same GFI and LFI), so one Content Object may be replicated and forwarded to multiple reverse paths corresponding to multiple pending Interests. The PIT tracks the interfaces out of which an Interest has been sent and ensures that similar Interests are not sent multiple times out the same interfaces. The PIT also ensures that similar Interests can flow in all directions. A forwarder, for example, with three interfaces 1, 2 and 3, may forward an interest received from interface 1 toward interfaces 2 and 3. At a later time, it receives a similar Interest from Interface 2. It may forward that Interest out of interface 1, but not 3.

[0039] The Content Store (CS) is an optional component. It stores recently seen or high-value Content Objects so later requests for the same object can be answered without forwarding an Interest. Cache policy and retention policy can be applied.

[0040] The Forwarding Information Base (FIB) contains information indicating the Interest forwarding routes. Typically, a routing protocol is used to populate the FIB. In one embodiment, the entries in the FIB are indexed based on the GFI or LFI.

[0041] In general, a forwarder matches the CI, GFI, and LFI of an Interest on the return path of a Content Object. This is because a malicious user could put in an CI for /popular/content and a GFI/LFI for a /colluding/site,

for example. The content object form /colluding/site would have malicious content, but the CI for /popular/content would be benign. If forwarders do not validate that the Content Object matches the full pending Interest with CI, GFI, and LFI, and only reverse-path forwards based on the CI, the malicious content could pollute the network.

[0042] To summarize the behavior of forwarding, an Interest is forwarded based on its GFI and LFI. When an intermediate node receives a Content Object, it verifies that it came from the expected direction, based on the PIT entry and the Content Object's CI, GFI, and LFI, which are expected to be identical with those of the Interest.

[0043] A PIT entry stores the CI and the ports to which the corresponding Content Object should be forwarded. Also within the same PIT entry are the associated GFI and LFI. It is possible that the PIT entry stores multiple GFI/LFI pairs for the same CI.

[0044] During operation, when a node creates an Interest, the node encapsulates the Interest in an express header. It computes the CI and places it in the header's CI field. If the node has knowledge of the proper GFI and LFI, it places them in their respective fields. The node then sends the Interest packet to the next-hop forwarder.

[0045] When a forwarder receives an Interest on an ingress interface, it performs the following actions: The forwarder looks up the CI/GFI/LFI in the PIT. If no entry exists, it creates a PIT entry for the Interest, and then proceeds to check the Content Store. To create a PIT entry, the forwarder records the CI, GFI, and LFI of the Interest and notes the ingress port on which the Interest is received. If the remaining time of the PIT entry is less than the Interest's requested holdtime, the forwarder can extend the PIT entry's remaining time. Note that the holdtime is a suggested maximum time to hold the Interest in a PIT. The forwarder then proceeds to forward the Interest.

[0046] If a forwarder implements a Content Store, it can look up the CI/GFI/LFI combination in the Content Store. If there is an exact match, the forwarder returns the Content Object and consumes the PIT entry. The returned object carries the same CI, GFI, and LFI. If there is no exact match in the Content Store, the forwarder forwards the Interest.

[0047] To forward the Interest, the forwarder first looks up the GFI in the FIB and finds the match in the FIB, and then forwards the Interest out those ports. If the FIB entry indicates that the forwarder is within the Interest's destination network domain, a local FIB can be used to further match the LFI. The forwarder is precluded from forwarding the Interest on the port from which it is received.

[0048] If an end-system content producer receives an Interest, it may create a Content Object that satisfies the body of the Interest and return it along the reverse path. The returned object carries the CI, GFI, and LFI received in the Interest. An end system may verify that the CI is properly calculated to match the body of the Interest.

[0049] An intermediate forwarder receiving a Content

Object first verifies whether the CI, GFI, and LFI of the received Content Object are in the PIT. If they are not, the forwarder drops the Content Object. The forwarder then verifies that the Content Object arrived from a port over which a corresponding Interest was previously forwarded, or over which the corresponding Interest could have been forwarded. If this condition is not met, the forwarder drops the Content Object.

[0050] If the forwarder implements a Content Store, the forwarder adds the object to the store if the object's holdtime permits it. Then the forwarder forwards the object along the reverse path. Subsequently, the forwarder consumes the PIT entries satisfied by the Content Object.

[0051] An end system receiving a Content Object should verify that the Content Object actually satisfies the original Interest. It should also verify the integrity of the Content Object's hash and signature.

[0052] FIG. 2 illustrates an exemplary express header for a CCN packet, in accordance with an embodiment of the present invention. In this example, a CCN packet 200 includes a payload portion 208 and an express header 210, which in turn can include a CI field 202, a GFI field 204, and an LFI field 206. Payload portion 208 may include the full CCN content name (i.e., HSVLI), and additional fields associated with an Interest or Content Object.

[0053] In one embodiment, the CI is only computed by the source node, and optionally verified by an authoritative source node generating content or responding from a long-term repository. The CI can be a hash of the full HSVLI based on, for example, the SHA-256 hashing algorithm. The CI can also be a hash of the Content Object requested.

[0054] FIG. 3 illustrates the process of forwarding an Interest and a corresponding Content Object, in accordance with an embodiment of the present invention. In this example, a client end system 300 is coupled to a network 300 and wishes to obtain content named "/parc/lab/ccn/document" from a content server 314. A number of forwarding nodes, such as forwarders 302, 304, 306, 308, 310, and 312, couple end system 300 with content server 314. In one embodiment, these forwarders can be IP routers. In this example, forwarders 302 and 312 can be enterprise gateway routers for the respective enterprise networks associated with end system 300 and content server 314. Forwarders 304 and 310 can be edge routers providing access into core network 300 to the enterprise networks.

[0055] During operation, end system 300 initiates an Interest for "/parc/lab/ccn/document". End system 300 computes the CI by hashing the entire HSVLI "/parc/lab/ccn/document," and optionally additional fields in the Interest. End system 300 also computes the GFI by hashing "/parc/lab" and the LFI by hashing "/parc/lab/ccn," for example. End system 300 then forwards the Interest to the default gateway router 302. Note that optionally end system 300 can query another node (for example a node providing a directory service) to ob-

tain the GFI and LFI. Alternatively, end system 300 can fill GFI and LFI with some default values and allow gateway router 302 to fill in these values. In turn, gateway router 302 and edge router 304 forwards the Interest based on the GFI. Similarly, core routers 306 and 308 can forward the Interest based on the GFI through core network 300 without having to parse the full HSVLI, using an exact match of the GFI in their respective FIB. When the Interest reaches edge router 310, edge router 310 forwards the Interest, based on the same GFI, to gateway router 312, which is within the enterprise network (i.e., the destination local network domain) where content server 314 resides. Since gateway router 312 has the routing information for the destination end system which is content server 314, gateway router 312 forwards the Interest based on the LFI to content server 314. Note that at each hop a corresponding entry is created at a local PIT. Each entry will contain the CI, GFI, and LFI.

[0056] When content server 314 returns a Content Object, it sets the Content Object's CI, GFI, and LFI to be the same as those in the Interest. Subsequently, the Content Object is reverse-path forwarded back to end system 300. At each hop, the CI, GFI, and LFI of the Content Object are checked against those values saved in the PIT, and the port on which the Content Object is also checked against the port on which the Interest has been previously forwarded. This reverse-path forwarding mechanism ensures that the Content Object travels along the same data path on which the Interest has traveled, and hence can prevent any malicious entity from tampering with or spoofing the returned Content Object.

[0057] FIG. 4 presents a flow chart illustrating the process of receiving and forwarding an Interest with an express header, in accordance with an embodiment of the present invention. During operation, the system receives an Interest (operation 400). The system then determines whether the CI of the Interest is in the PIT (operation 402). If so, the system adds the ingress port on which the Interest is received to the corresponding PIT entry (operation 404). Otherwise, the system further determines whether the content corresponding to the CI is in the local Content Store (operation 406). If so, the system returns the matching Content Object (operation 408). If not, the system then adds a corresponding entry to the PIT (operation 407).

[0058] Next, the system determines whether the Interest's GFI is in the FIB (operation 410). If the GFI is not in the FIB, the system drops the Interest packet (operation 412). If the GFI is in the FIB, the system further determines whether the local forwarder is in the same destination network domain as indicated by the GFI (operation 414). If not, the system forwards the Interest packet based on the egress port indicated by the FIB entry corresponding to the GFI (operation 416). If the FIB entry indicates that the forwarder is within the same destination network domain as indicated by the GFI, the system then looks up the LFI in the FIB and determines the egress

port which can be used to forward the Interest to the end system hosting the Content Object (operation 416). Note that the forwarding information associated with the LFI can be stored in the same FIB for GFI lookup, or a separate, local FIB dedicated to storing LFI forwarding information. The system then forward the Interest packet accordingly (operation 420).

[0059] FIG. 5 illustrates an exemplary forwarding information base (FIB), in accordance with an embodiment of the present invention. In this example, the FIB includes a global FIB 500 and a local FIB 520. Global FIB includes a GFI column 502, an egress port(s) column 504, and a local FIB indication column 506. Local FIB includes an LFI column 522 and an egress port column 524.

[0060] GFI column 502 stores the GFIs for which global FIB 500 maintains the proper forwarding (i.e., egress port(s)) information. Egress port(s) column 504 indicates one or more egress ports via which an Interest packet can be forwarded based on its GFI.

[0061] Local FIB indication column 506 stores an indicator which indicates whether the local forwarder is within the same network domain as indicated by an Interest's GFI. If the forwarder is in the same domain, the system will further look up the LFI in local FIB 520, in which case the corresponding egress port entry in column 504 may contain a null value.

[0062] LFI column 522 in local FIB 520 stores the LFI for a locally routable Interest, and the corresponding egress port information is stored in egress port column 524.

[0063] FIG. 6 illustrates an exemplary pending Interest table (PIT), in accordance with an embodiment of the present invention. In this example, a PIT 600 includes a CI column 602, an ingress port(s) column 604, a GFI column 606, and an LFI column 608. CI column 602 stores the CI for a pending Interest and is used to look up an pending Interest in PIT 600. Ingress port(s) column 604 indicates one or more ingress ports on which an Interest is received. These ports will be used to send back the Content Objects corresponding to the pending Interest. GFI column 606 and LFI column 608 indicate the GFI and LFI, respectively, which a corresponding received Content Object should have, and are used to confirm that the Content Object is received via the correct reverse path.

[0064] FIG. 7 presents a flow chart illustrating an exemplary process of receiving and forwarding a Content Object with an express header, in accordance with an embodiment of the present invention. During operation, the system first receives a Content Object packet (operation 702). The system then determines whether the CI of the Content Object is in the PIT (operation 704). If it is not in the PIT, the system discards the packet (operation 706). Otherwise, the system further determines whether the GFI and LFI in the Content Object match the corresponding PIT entry (corresponding to GFI column 606 and LFI column 608 in FIG. 6) (operation 708). If not, the system discards the packet (operation 706). Other-

wise, the system determines whether the ingress port on which the Content Object is received matches the FIB entry corresponding to the GFI or LFI (operation 712). If not, the system discards the packet (operation 706). Otherwise, the system forwards the Content Object packet to the ports indicated by the PIT entry (corresponding to ingress port(s) column 604 in FIG. 6) (operation 714).

[0065] FIG. 8 illustrates an exemplary system for forwarding packets with HSVLI and express headers, in accordance with an embodiment. A system 800 for forwarding packets with HSVLI and express headers comprises a processor 810, a memory 820, and a storage device 830. Storage device 830 typically stores instructions which can be loaded into memory 820 and executed by processor 810 to perform the hash-forwarding methods mentioned above. In one embodiment, the instructions in storage 830 can implement an express header module 832, a PIT module 834, a FIB module 836, and a CS module 838 all of which can be in communication with each other through various means. Express header module 832 can compute the CS, GFI, and LFI for an interest.

[0066] In some embodiments, modules 832, 834, 836, and 838 can be partially or entirely implemented in hardware and can be part of processor 810. Further, in some embodiments, the system may not include a separate processor and memory. Instead, in addition to performing their specific tasks, modules 832, 834, 836, and 838, either separately or in concert, may be part of general- or special-purpose computation engines.

[0067] Storage device 830 stores programs to be executed by processor 810. Specifically, storage 830 stores a program that implements a system (application) for performing forwarding of packets with HSVLI and express headers. During operation, the application program can be loaded from storage 830 into memory 820 and executed by processor 810. As a result, system 800 can perform the functions described above. System 800 can be coupled to an optional display 880, keyboard 860, and pointing device 870, and also be coupled via one or more network interfaces to network 882.

[0068] The data structures and code described in this detailed description are typically stored on a computer-readable storage medium, which may be any device or medium that can store code and/or data for use by a computer system. The computer-readable storage medium includes, but is not limited to, volatile memory, non-volatile memory, magnetic and optical storage devices such as disk drives, magnetic tape, CDs (compact discs), DVDs (digital versatile discs or digital video discs), or other media capable of storing computer-readable media now known or later developed.

[0069] The methods and processes described in the detailed description section can be embodied as code and/or data, which can be stored in a computer-readable storage medium as described above. When a computer system reads and executes the code and/or data stored on the computer-readable storage medium, the compu-

ter system performs the methods and processes embodied as data structures and code and stored within the computer-readable storage medium.

[0070] Furthermore, methods and processes described herein can be included in hardware modules or apparatus. These modules or apparatus may include, but are not limited to, an application-specific integrated circuit (ASIC) chip, a field-programmable gate array (FPGA), a dedicated or shared processor that executes a particular software module or a piece of code at a particular time, and/or other programmable-logic devices now known or later developed. When the hardware modules or apparatus are activated, they perform the methods and processes included within them.

[0071] The above description is presented to enable any person skilled in the art to make and use the embodiments, and is provided in the context of a particular application and its requirements. Various modifications to the disclosed embodiments will be readily apparent to those skilled in the art, and the general principles defined herein may be applied to other embodiments and applications without departing from the spirit and scope of the present disclosure. Thus, the present invention is not limited to the embodiments shown, but is to be accorded the widest scope consistent with the principles and features disclosed herein.

Claims

1. A computer-executable method for forwarding named packets, the method comprising:

receiving a named packet with an express header, wherein the named packet includes a payload and a name which identifies the payload; and wherein the express header includes a content identifier, a global forwarding identifier, and a local forwarding identifier; wherein the content identifier uniquely represents an interest corresponding to the name; wherein the global forwarding identifier includes forwarding information to facilitate global forwarding; and wherein the local forwarding identifier includes forwarding information to facilitate local forwarding; and making a forwarding decision for the packet based on the content identifier, the global forwarding identifier, and optionally the local forwarding identifier.

2. The method of claim 1, wherein the name is one of:

a hierarchically structured variable-length identifier (HSVLI),
a set of attributes,
a flat identifier,

an array of bytes,
 an entire set of data, and
 a portion of the data.

3. The method of claim 1, wherein the packet is an interest in a piece of content corresponding to the name;
 wherein the content identifier is a hash derived based on the entire name and optionally additional information in the packet; and
 wherein the global forwarding identifier is a hash derived on the subset of the name components. 5
 10
4. The method of claim 1, further comprising obtaining the global forwarding identifier from a node providing a directory service. 15
5. The method of claim 1, further comprising computing the global forwarding identifier based on the name or content of the packet. 20
6. The method of claim 1, further comprising computing the local forwarding identifier based on the name or content of the packet. 25
7. The method of claim 1, wherein the name is an HSVLI, and
 wherein the method further comprises computing the global forwarding identifier by hashing a predetermined number of components of the HSVLI or a number of components of the HSVLI based on each component's type. 30
8. The method of claim 1, wherein the name is an HSVLI, and
 wherein the method further comprises computing the local forwarding identifier by hashing a predetermined number of components of the HSVLI or a number of components of the HSVLI based on each component's type. 35
 40
9. The method of claim 1, further comprising maintaining a pending interest table, wherein a respective entry in the pending interest table indicates a pending interest associated with the content identifier, global forwarding identifier, and local forwarding identifier. 45
10. The method of claim 1, further comprising maintaining a forwarding information base, wherein a respective entry in the forwarding information base indicates forwarding information for a packet that contains a global forwarding identifier. 50

55

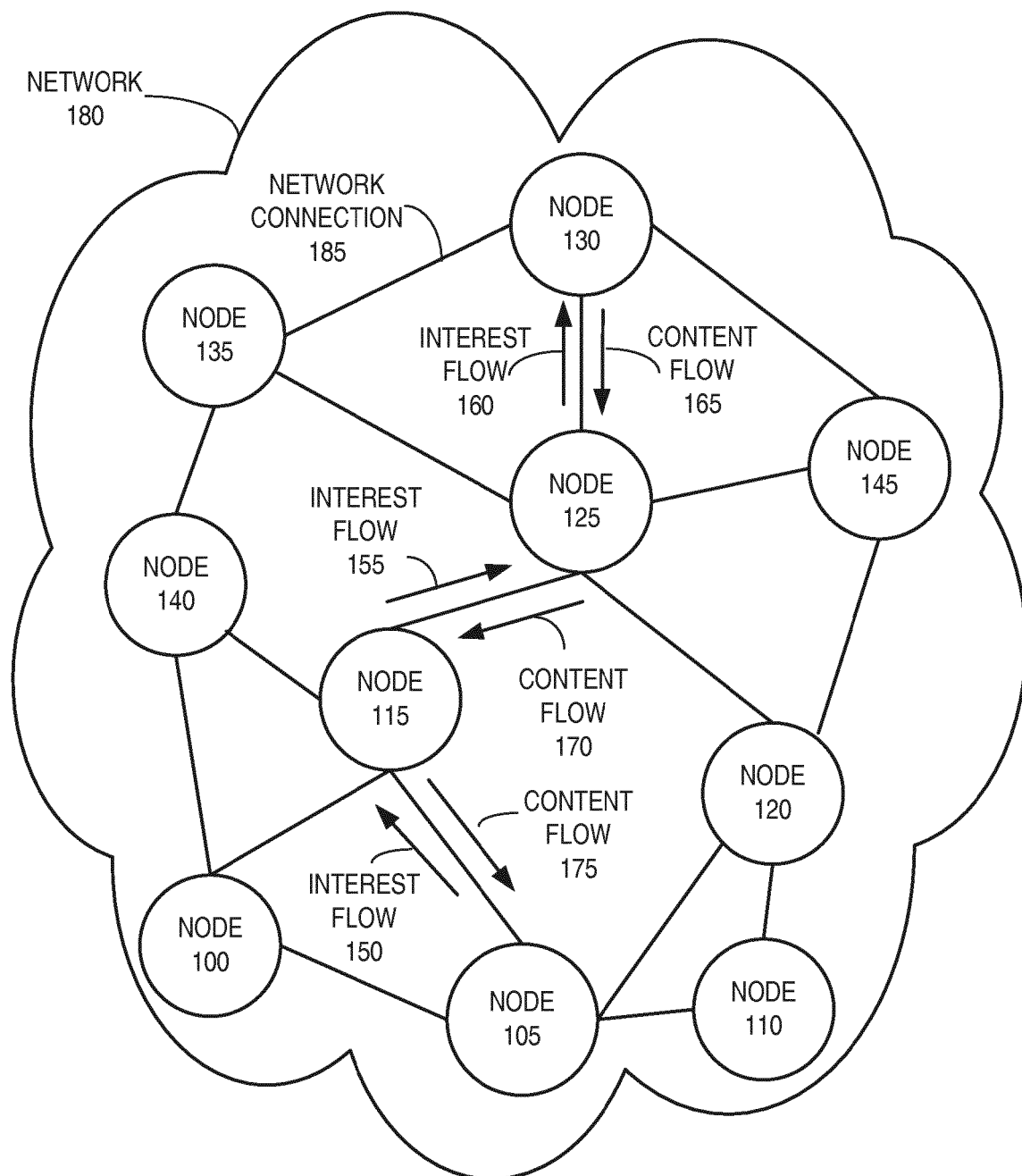


FIG. 1

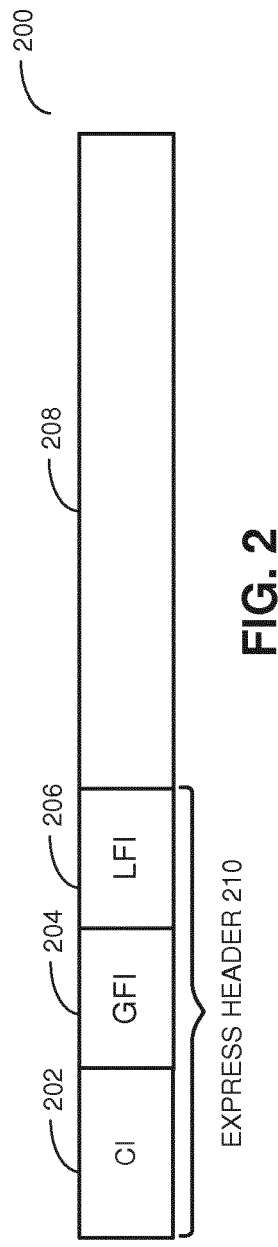


FIG. 2

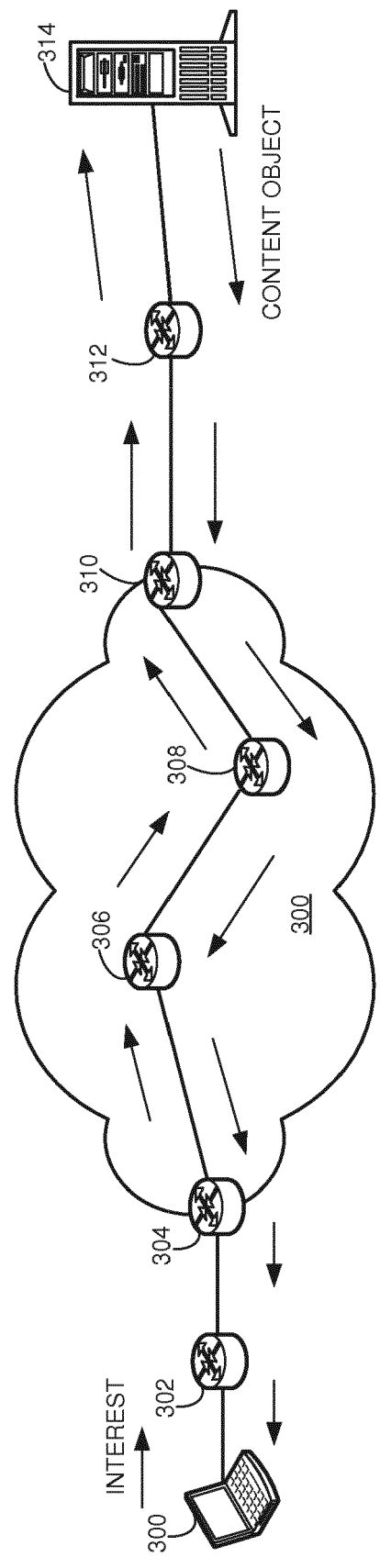


FIG. 3

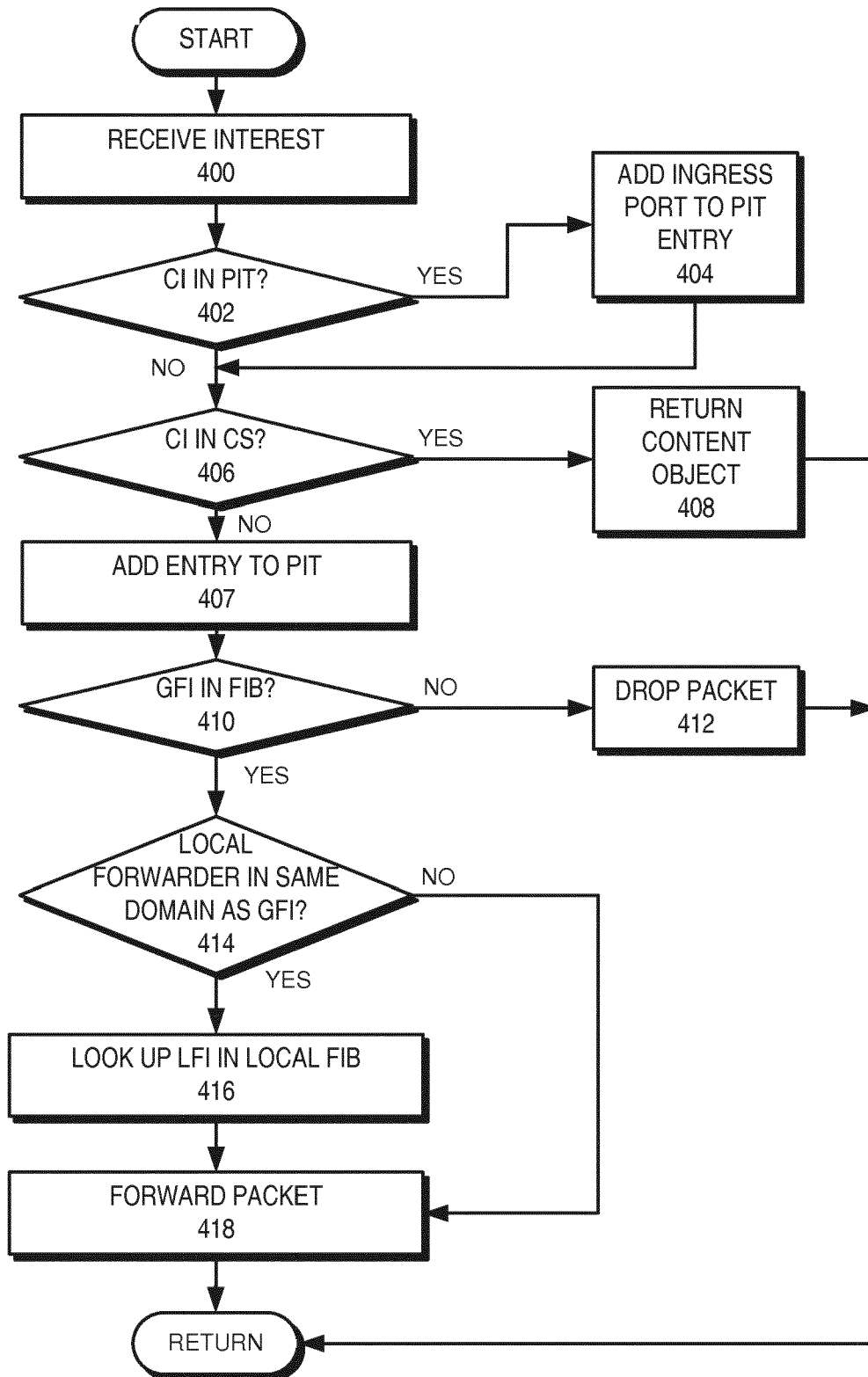


FIG. 4

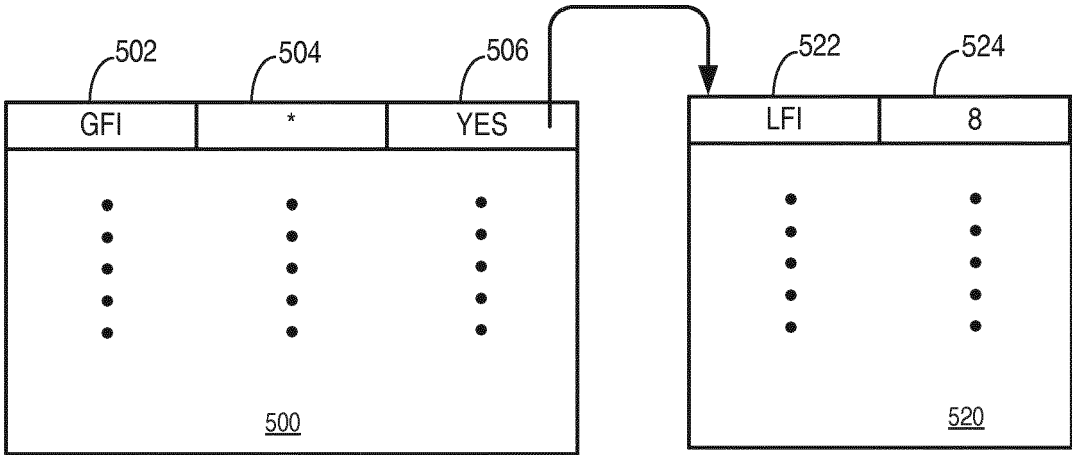


FIG. 5

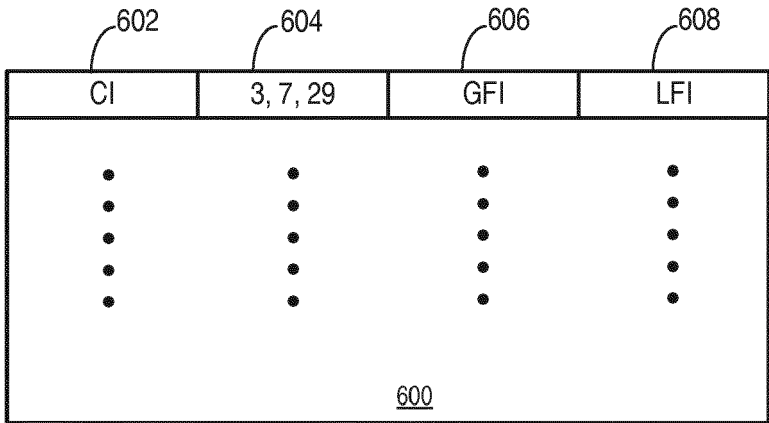


FIG. 6

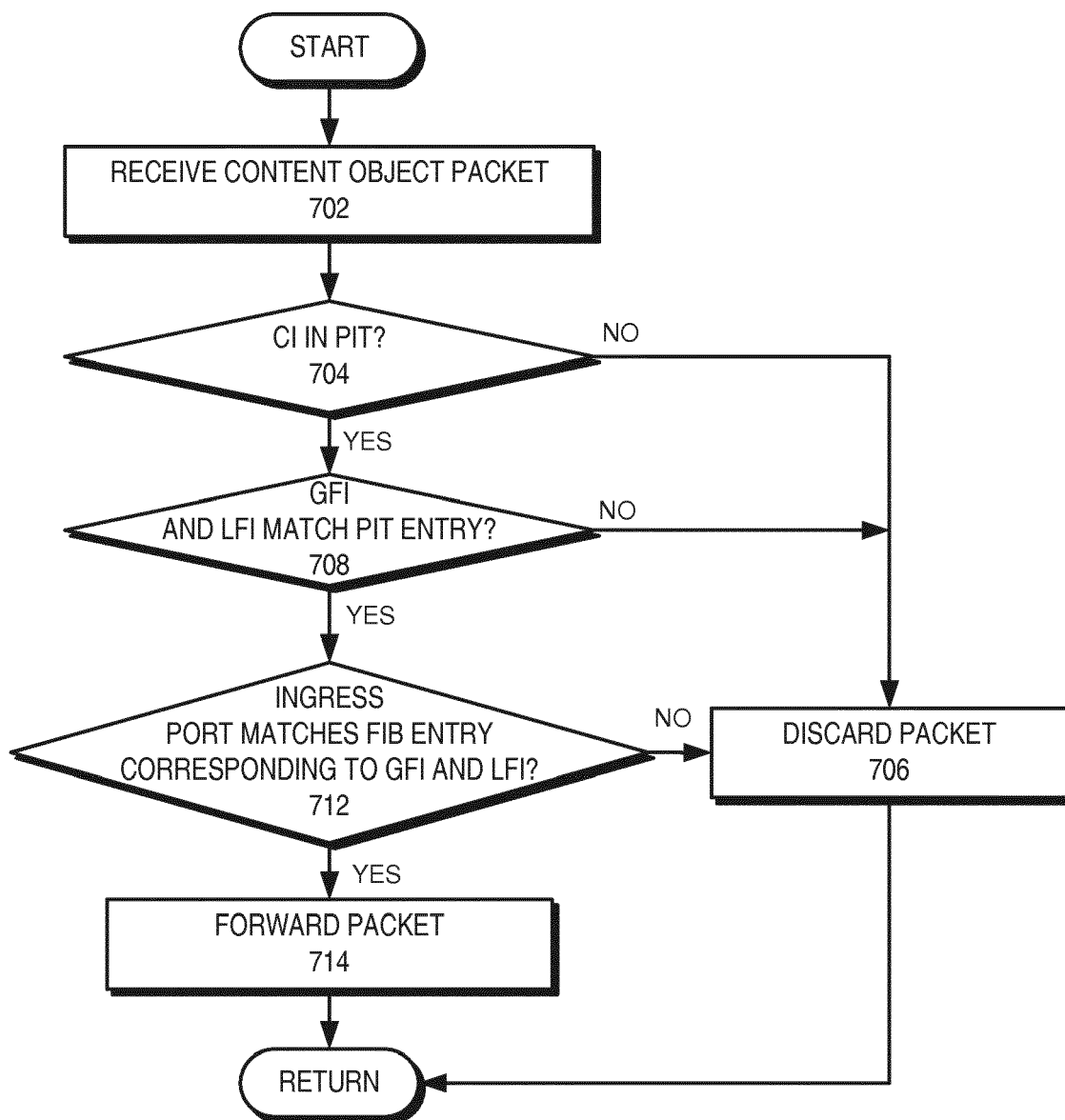


FIG. 7

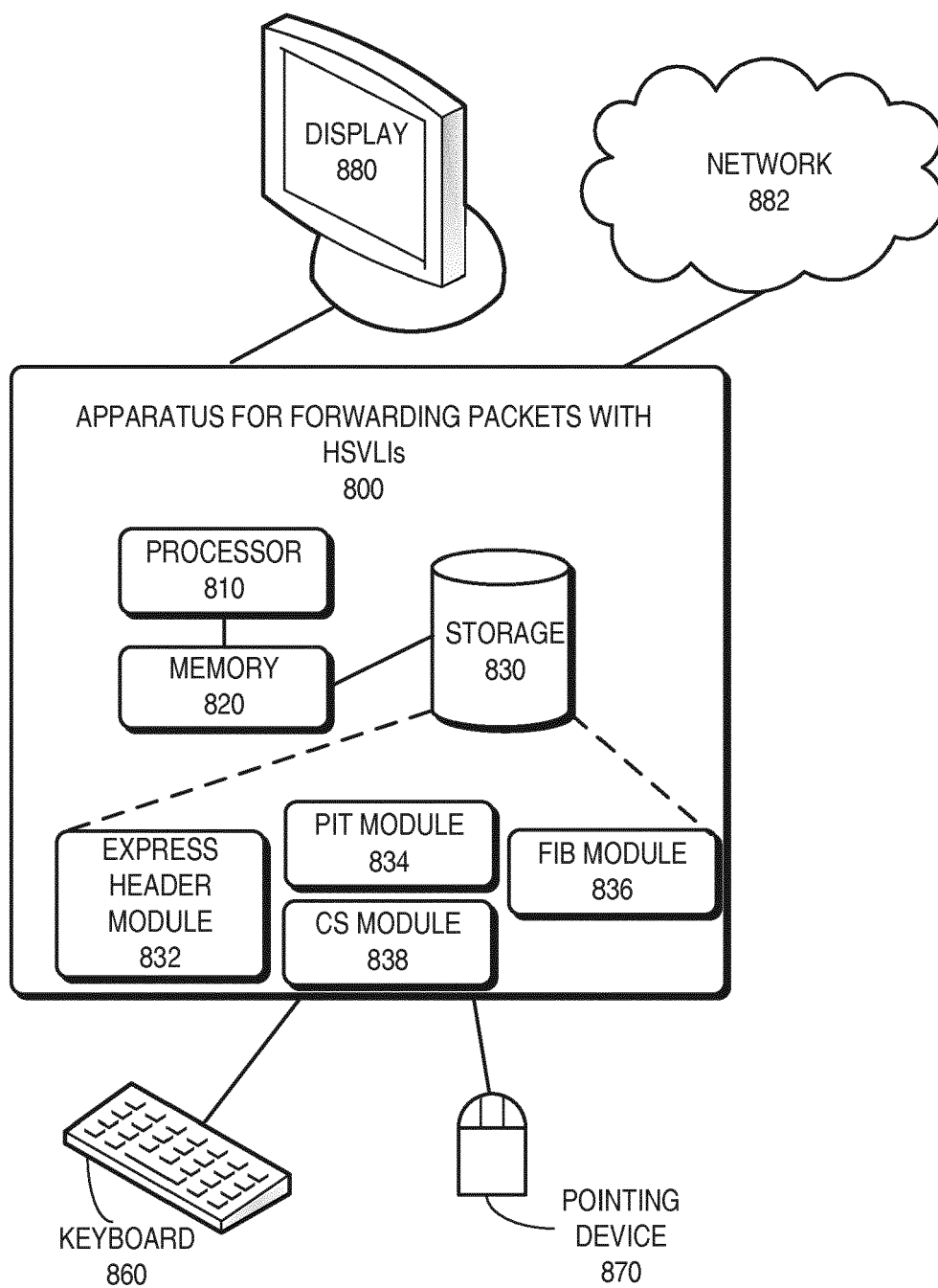


FIG. 8



EUROPEAN SEARCH REPORT

Application Number
EP 14 18 9968

5

10

15

20

25

30

35

40

45

50

55

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (IPC)
A	EP 2 654 269 A1 (SAMSUNG ELECTRONICS CO LTD [KR]) 23 October 2013 (2013-10-23) * paragraphs [0019] - [0022] * * figure 3 *	1-10	INV. H04L12/743 H04L12/745 H04L12/701
A	"Identification framework in future networks; Y.3031 (05/12)", ITU-T STANDARD, INTERNATIONAL TELECOMMUNICATION UNION, GENEVA ; CH, no. Y.3031 (05/12), 7 May 2012 (2012-05-07), page 1, XP017577703, [retrieved on 2012-11-27] * page 5, paragraph 7.2 *	1-10	
A	EP 2 214 356 A1 (PALO ALTO RES CT INC [US]) 4 August 2010 (2010-08-04) * figures 3-5 *	1-10	
A	US 2011/271007 A1 (WANG GUO QIANG [US] ET AL) 3 November 2011 (2011-11-03) * paragraph [0140] - paragraph [0149] * * figures 5,6 *	1-10	TECHNICAL FIELDS SEARCHED (IPC) H04L
A	TORU HASEGAWA: "A Survey of the Research on Future Internet and Network Architectures", IEICE TRANSACTIONS ON COMMUNICATIONS, COMMUNICATIONS SOCIETY, TOKYO, JP, vol. E96B, no. 6, 1 June 2013 (2013-06-01), pages 1385-1401, XP001584385, ISSN: 0916-8516, DOI: HTTP://DX.DOI.ORG/10.1587/TRANSCOM.E96.B.1385 * page 1387, paragraph 3.3 - page 1388, paragraph 3.3.2 *	1-10	
The present search report has been drawn up for all claims			
Place of search Berlin		Date of completion of the search 24 February 2015	Examiner Siebel, Christian
CATEGORY OF CITED DOCUMENTS X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons & : member of the same patent family, corresponding document			

EPO FORM 1503 03.82 (P04C01)



EUROPEAN SEARCH REPORT

Application Number
EP 14 18 9968

5

10

15

20

25

30

35

40

45

50

55

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (IPC)
A	EP 2 323 346 A1 (PALO ALTO RES CT INC [US]) 18 May 2011 (2011-05-18) * paragraphs [0002] - [0008] * * page 25 - page 33 * -----	1-10	
			TECHNICAL FIELDS SEARCHED (IPC)
The present search report has been drawn up for all claims			
Place of search Berlin		Date of completion of the search 24 February 2015	Examiner Siebel, Christian
CATEGORY OF CITED DOCUMENTS X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons & : member of the same patent family, corresponding document			

EPO FORM 1503 03.82 (P04C01)

**ANNEX TO THE EUROPEAN SEARCH REPORT
ON EUROPEAN PATENT APPLICATION NO.**

EP 14 18 9968

5

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report. The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

24-02-2015

10

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 2654269 A1	23-10-2013	NONE	
EP 2214356 A1	04-08-2010	CN 101795229 A	04-08-2010
		EP 2214356 A1	04-08-2010
		JP 5525273 B2	18-06-2014
		JP 2010178342 A	12-08-2010
		KR 20100088561 A	09-08-2010
		US 2010195653 A1	05-08-2010
US 2011271007 A1	03-11-2011	US 2011271007 A1	03-11-2011
		US 2013322451 A1	05-12-2013
EP 2323346 A1	18-05-2011	CN 102045252 A	04-05-2011
		EP 2323346 A1	18-05-2011
		JP 2011091802 A	06-05-2011
		KR 20110043476 A	27-04-2011
		US 2011090908 A1	21-04-2011

30

35

40

45

50

55

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82